

WHAT IS CLAIMED IS:

1. A system comprising:

5 a distributed store comprising a primary state of session data configured for access
by a plurality of application servers, wherein the session data comprises
one or more attributes;

10 a first one of the application servers comprising a client state of the session data,
wherein the application server is configured to provide access to the
session data to processes executing within the first application server;

wherein the first application server is configured to:

15 track mutable accesses of the attributes in the client state;

20 perform an object graph comparison of mutably accessed attributes with a
benchmark version of the client state to determine a subset of
modified attributes, wherein the benchmark version of the client
state comprises a previous version of the attributes in the client
state; and

25 wherein the distributed store is configured to synchronize the primary state with
the client state according to the subset of modified attributes.

2. The system as recited in claim 1, wherein a mutable access comprises a
write access to an attribute.

30 3. The system as recited in claim 1, wherein, to synchronize the primary state
with the client state, the distributed store is further configured to update the primary state

using the subset of modified attributes.

4. The system as recited in claim 1, wherein the distributed store is further configured to lock the primary state for access by a process executing on the first application server, wherein, while the primary state is locked for the process, other processes on the application server cannot access the primary state.

5. The system as recited in claim 4, wherein the distributed store is further configured to lock the primary state for access by a process executing on the first application server, wherein, while the primary state is locked for the process, other processes on the application server or on the other application servers cannot access the primary state.

6. The system as recited in claim 4, wherein the distributed store is further configured to request the process to release the lock, wherein the process is configured to release the lock in response to said request.

7. The system as recited in claim 4, wherein the process is configured to release the lock when locked access to the primary state is no longer required by the process.

8. The system as recited in claim 4, wherein the distributed store is further configured to lock at least a portion of the primary state for access by a thread executing within the process, wherein, while the at least a portion of the primary state is locked for the thread, other threads executing within the process cannot access the at least a portion of the primary state.

9. The system as recited in claim 1, wherein the primary state is distributed across a plurality of devices.

30

10. The system as recited in claim 1 further comprising another instance of the primary state, wherein the system is configured to:

determine differences between the primary state and a benchmark of the primary state; and

synchronize the other instance of the primary state with the primary state according to the determined differences.

11. The system as recited in claim 10, wherein to determine differences between the primary state and a benchmark version of the primary state, the system is further configured to perform a binary comparison of the primary state and the benchmark version of the primary state.

12. The system as recited in claim 10, wherein to determine differences between the primary state and a benchmark version of the primary state, the system is further configured to perform an object graph comparison of the primary state and the benchmark version of the primary state.

13. The system as recited in claim 1, wherein the first application server is further configured to:

track accesses of mutable attributes of the client state;

perform an object graph comparison of accessed mutable attributes and the benchmark version of the session data to determine a subset of modified mutable attributes; and

wherein, to synchronize the primary state with the client state, the distributed store is further configured to use the subset of modified mutable attributes.

14. A system, comprising:

5 a plurality of application servers, wherein each of the plurality of application servers comprises a client state of session data comprising one or more attributes, wherein each of the application servers is configured to provide access to the session data to processes executing within the particular application server;

10 a distributed store comprising a primary state of the session data configured for access by the plurality of application servers;

wherein each of the plurality of application servers is configured to:

15 track mutable accesses of the attributes in the client state of the particular application server;

perform an object graph comparison of mutably accessed attributes with a benchmark version of the client state to determine a subset of modified attributes, wherein the benchmark version of the client state comprises a previous version of the attributes in the client state;

20

25 wherein the distributed store is configured to synchronize the primary state with the client state according to the subset of modified attributes.

15. The system as recited in claim 14, wherein a mutable access comprises a write access to an attribute.

30 16. The system as recited in claim 14, wherein, to synchronize the primary

state with the client state, the distributed store is further configured to update the primary state with the subset of modified attributes.

17. The system as recited in claim 14, wherein the distributed store is further
5 configured to provide locked access to the primary state to processes executing on the plurality of application servers, wherein, while the primary state is locked for a process, other processes cannot access the primary state.

18. The system as recited in claim 17, wherein the distributed store is further
10 configured to request the process to release the lock, wherein the process is configured to release the lock in response to said request.

19. The system as recited in claim 17, wherein the process is configured to
15 release the lock when locked access to the primary state is no longer required by the process.

20. The system as recited in claim 17, wherein the distributed store is further
configured to lock at least a portion of the primary state for access by a thread executing within the process, wherein, while the at least a portion of the primary state is locked for
20 the thread, other threads executing within the process cannot access the at least a portion of the primary state.

21. The system as recited in claim 14, wherein the primary state is distributed
25 across a plurality of devices.

22. The system as recited in claim 14, further comprising another instance of
the primary state, wherein the system is configured to:

determine differences between the primary state and a benchmark version of the
30 primary state; and

synchronize the other instance of the primary state with the primary state using the determined differences.

5 23. The system as recited in claim 22, wherein to determine differences between the primary state and a benchmark version of the primary state, the system is further configured to perform a binary comparison of the primary state and the benchmark version of the primary state.

10 24. The system as recited in claim 22, wherein to determine differences between the primary state and a benchmark of the primary state, the system is further configured to perform an object graph comparison of the primary state and the benchmark version of the primary state.

15 25. The system as recited in claim 14, wherein each of the plurality of application servers is further configured to:

track accesses of mutable attributes in the client state;

20 perform an object graph comparison of the accessed mutable attributes and the benchmark version of the session data to determine a subset of modified mutable attributes; and

25 wherein, to synchronize the primary state with the client state, the distributed store is further configured to use the subset of modified mutable attributes.

26. A system, comprising:

30 means to lock access to a primary state of session data configured for access by a plurality of application servers for a process executing on one of the

plurality of application servers, wherein the session data comprises a plurality of attributes;

wherein, while the primary state is locked for the process, other processes cannot access the primary state;

wherein each of the plurality of application servers comprises a client state of the session data accessible to processes executing within the application server;

means for each of the application servers to:

determine a set of mutably accessed attributes of the client state of the particular application server;

determine a subset of the set of mutably accessed attributes that are modified in respect to the primary state; and

means to synchronize the primary state with the client state using the subset of modified attributes.

27. The system as recited in claim 26, wherein said means for each of the application servers to determine a set of mutably accessed attributes of the client state of the particular application server comprises tracking mutable accesses of the attributes in the client state.

28. The system as recited in claim 26, wherein said means for each of the application servers to determine a subset of the set of mutably accessed attributes that are modified in respect to the primary state comprises performing an object graph comparison of the mutably accessed attributes with a benchmark version of the client state to

determine the subset of modified attributes, wherein the benchmark version of the client state comprises a previous version of the attributes in the client state.

5 29. A method, comprising:

tracking mutable accesses of a plurality of attributes of a client state of session data, wherein the client state is associated with an application server;

10 performing an object graph comparison of mutably accessed attributes with a benchmark version of the client state to determine a subset of modified attributes, wherein the benchmark version of the client state comprises a previous version of the attributes in the client state; and

15 synchronizing a primary state of the session data with the client state according to the subset of modified attributes, wherein the primary state is accessible by a plurality of application servers.

20 30. The method as recited in claim 29, wherein a mutable access comprises a write access to an attribute.

31. The method as recited in claim 29, wherein said synchronizing comprises updating the primary state using the subset of modified attributes.

25 32. The method as recited in claim 29, further comprising locking the primary state for access by a process executing on the application server, wherein, while the primary state is locked for access by the process, other processes on the plurality of application servers cannot access the primary state.

30 33. The method as recited in claim 32, further comprising:

the process receiving a request to release locked access to the primary state; and

the process releasing the locked access to the primary state in response to said request.

5

34. The method as recited in claim 32, further comprising releasing locked access to the primary state when no longer required by the process.

35. The method as recited in claim 32, further comprising locking one or more
10 attributes of the primary state for access by a thread executing within the process, wherein, while the one or more attributes are locked for access by the thread, other threads executing within the process cannot access the one or more attributes.

36. The method as recited in claim 29, further comprising:
15 determining differences between the primary state and a benchmark version of the primary state; and

synchronizing another instance of the primary state with the primary state using
20 the determined differences.

37. The method as recited in claim 36, wherein said determining differences
between the primary state and a benchmark version of the primary state comprises
performing a binary comparison of the primary state and the benchmark version of the
25 primary state.

38. The method as recited in claim 36, wherein said determining differences
between the primary state and a benchmark version of the primary state comprises
performing an object graph comparison of the primary state and the benchmark version of
30 the primary state.

39. The method as recited in claim 29, further comprising:

tracking accesses of mutable attributes of the client state;

5

performing an object graph comparison of accessed mutable attributes and the benchmark version of the session data to determine a subset of modified mutable attributes; and

10

using the subset of modified mutable attributes in said synchronizing the primary state with the client state.

40. An article of manufacture comprising software instructions executable to implement:

15

tracking mutable accesses of a plurality of attributes of a client state of session data, wherein the client state is associated with an application server;

20

performing an object graph comparison of mutably accessed attributes with a benchmark version of the client state to determine a subset of modified attributes, wherein the benchmark version of the client state comprises a previous version of the attributes in the client state; and

25

synchronizing a primary state of the session data with the client state using the subset of modified attributes, wherein the primary state is accessible by a plurality of application servers.

30

41. The article of manufacture as recited in claim 40, wherein a mutable access comprises a write access to an attribute.

42. The article of manufacture as recited in claim 40, wherein, in said synchronizing, the software instructions are further executable to implement updating the primary state using the subset of modified attributes.

43. The article of manufacture as recited in claim 40, wherein the software instructions are further executable to implement locking the primary state for access by a process executing on the application server, wherein, while the primary state is locked for access by the process, other processes on the plurality of application servers cannot access the primary state.

44. The article of manufacture as recited in claim 43, wherein the software instructions are further executable to implement:

the process receiving a request to release locked access to the primary state; and

the process releasing the locked access to the primary state in response to said request.

45. The article of manufacture as recited in claim 43, wherein the software instructions are further executable to implement releasing locked access to the primary state when no longer required by the process.

46. The article of manufacture as recited in claim 43, wherein the software instructions are further executable to implement locking one or more attributes of the primary state for access by a thread executing within the process, wherein, while the one or more attributes are locked for access by the thread, other threads executing within the process cannot access the one or more attributes.

47. The article of manufacture as recited in claim 40, wherein the software instructions are further executable to implement:

determining differences between the primary state and a benchmark version of the
primary state; and

5 synchronizing another instance of the primary state with the primary state using
the determined differences.

48. The article of manufacture as recited in claim 47, wherein, in said
determining differences between the primary state and a benchmark version of the
10 primary state, the software instructions are further executable to implement a binary
comparison of the primary state and the benchmark version of the primary state.

49. The article of manufacture as recited in claim 47, wherein, in said
determining differences between the primary state and a benchmark version of the
15 primary state, the software instructions are further executable to implement an object
graph comparison of the primary state and the benchmark version of the primary state.

50. The article of manufacture as recited in claim 40, wherein the software
instructions are further executable to implement:

20 tracking accesses of mutable attributes of the client state;

performing an object graph comparison of accessed mutable attributes and the
benchmark version of the session data to determine a subset of modified
25 mutable attributes; and

wherein the software instructions are further executable to implement using the
subset of modified mutable attributes in said synchronizing the primary
state with the client state.

30